

# **GUI DEVELOPMENT FOR DC MOTOR APPLICATION IN MATLAB**

NAZARIHA BINTI MAT SALLEH

This thesis is submitted as partial fulfillment of the requirements for the award of the  
Bachelor of Electrical Engineering (Hons.) (Electronics)

Faculty of Electrical & Electronics Engineering  
Universiti Malaysia Pahang

APRIL, 2009

“All the trademark and copyrights use herein are property of their respective owner. References of information from other sources are quoted accordingly; otherwise the information presented in this report is solely work of the author.”

Signature : \_\_\_\_\_

Author : NAZARIHA BINTI MAT SALLEH

Date : 27 APRIL 2009

To my beloved mother and father:

Mrs. Che Meryam Mohammed and Mr Mat Salleh Noh

My sibling:

Rohani Mat Salleh, Rohana Mat Salleh, Mohd Al Amin Mat Salleh and my little brother

Muhammad Khairul Anwar Mat Salleh

Also to my all niece and nephew.

## **ACKNOWLEDGEMENT**

Alhamdulillah, the highest thank to God because with His Willingness I possible to complete the final year project. In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my main thesis supervisor, Mr. Reza Ezuan bin Samin, for encouragement, guidance, critics and friendship.

I would also like to thank to all UMP lecturers and electrical technicians whom had helped directly or indirectly in what so ever manner thus making this project a reality.

My special thanks for my parents, Mr. Mat Salleh bin Noh and Mrs Che Meryam bt Mohammed and also the rest of my family, for their financial, spiritual support and pray on me throughout this project. Their blessing gave me the high-spirit and strength to face any problem occurred and to overcome them rightly.

The episode of acknowledgement would not be complete without the mention of my fellow colleagues in 4BEE, 2008/09 session. Finally, I would like to thank all whose direct and indirect support helped me completing my thesis in time. Only Allah can repay your kindness.

## **ABSTRACT**

The application of the DC motor in real life is very command. In industry, DC motor controls are very important to make the mechanical movement operation running. There are many applications that have been develop by using motor control in electronic field such as Computer Integral Manufacturing (CIM) and Flexible Manufacturing System (FMS). The objective of this project is to develop the Graphical User Interface of motor control and temperature sensor through MATLAB GUIDE, interface the MATLAB GUI that consist of transmitter and receiver program with hardware via serial communication and control the DC motor and temperature sensor. The PIC is used to control DC motor and temperature sensor. By using MATLAB GUIDE, the process of laying out and programming GUIs and interface with PIC via serial communication port to control the DC motor and temperature sensor will be easier because it is already provides a set of tools. As a result, the DC motor control and temperature sensor that consist of transmit and receiver program is able to be controlled by using MATLAB GUI and interface the MATLAB GUI with PIC via serial communication port.

## **ABSTRAK**

Penggunaan motor DC dalam kehidupan ialah suatu kebiasaan. Dalam industri, pengawalan motor DC sangat penting untuk pergerakan mekanikal dijalankan. Ia mempunyai banyak penggunaan yang dibangunkan dengan menggunakan kawalan motor dalam bidang elektronik seperti Computer Integral Manufacturing (CIM) dan Flexible Manufacturing System (FMS). Tujuan projek ini adalah untuk membina grafik antaramuka pengguna untuk mengawal motor DC dan pengesanan suhu melalui MATLAB, ia mengandungi penghantaran dan penerimaan data program dengan perkakasan elektrik melalui pintu komunikasi dan mengawal motor dc serta pengesanan suhu. PIC digunakan untuk mengawal motor dan pengesanan suhu. Dengan menggunakan MATLAB GUIDE, proses meletak dan membina GUIs dan menghubungkan dengan PIC melalui pintu komunikasi untuk mengawal motor dc dan pengesanan suhu dengan mudah kerana ia telah mempunyai set peralatan yang disediakan. Kesimpulannya, pengawalan motor DC dan pengesanan suhu yang mempunyai penghantaran dan penerimaan data program dengan senang dapat dikawal menggunakan MATLAB GUI dan menghubungkan MATLAB GUI kepada PIC melalui pintu komunikasi seri.

## TABLE OF CONTENTS

CHAPTER	SUBJECT	PAGE
	TITLE	i
	DECLARATION	ii
	DEDICATION	iii
	ACKNOWLEDGEMENT	iv
	ABSTRACT	v
	ABSTRAK	vi
	TABLE OF CONTENT	vii
	LIST OF FIGURE	x
	LIST OF TABLE	xii
 <b>1</b>	 <b>INTRODUCTION</b>	
	1.1 Overview	1
	1.2 Objective	3
	1.3 Scope of The Project	3
	1.4 Problem Statement	4
	1.5 Thesis Organization	5
 <b>2</b>	 <b>LITERATURE REVIEW</b>	
	2.1 PIC Microcontroller	6
	2.1.1 Origins	7

2.1.2	PIC Microcontroller Option	8
2.1.3	Variants	9
2.2	PIC Basic Pro Compiler	10
2.3	Graphical User Interface (GUI)	11
2.3.1	General Definition of GUI	11
2.3.2	MATLAB GUI	11
2.3.3	MATLAB GUIDE	12
2.3.3.1	Two Basic in Process of implementing a GUI	13
2.3.4	Operation of GUI	13
2.4	DC Motor	15
2.4.1	Introduction	15
2.4.2	The Advantage and The Drawbacks	16
2.4.3	Type of DC Motor	17
2.4.3.1	Stepper Motor	17
2.4.3.2	Coreless DC Motor	18
2.4.3.3	Brushless DC Motor	18
2.5	Temperature Sensor	19
2.6	Darlington Transistor C1815	21
2.7	Relay	21
2.8	MAX232	22
<b>3</b>	<b>METHODOLOGY</b>	
3.1	Introduction	24
3.2	Methodology	25
3.2.1	Hardware Installation	27
3.2.2	Development MATLAB GUI Using MATLAB GUIDE	31
3.2.3	Build MATLAB Programming	37
3.2.4	Build PIC Programming	42



<b>4</b>	<b>RESULT AND DISCUSSION</b>	
4.1	Introduction	46
4.2	Main Menu of the GUI	46
4.3	Interface MATLAB GUI Software	50
4.4	User Information GUI	54
<b>5</b>	<b>CONCLUSION AND RECOMMENDATION</b>	
5.1	Conclusion	56
5.2	Future Recommendation	57
5.3	Costing and Commercialization	57
5.3.1	Costing	57
5.3.2	Commercialization	59
	<b>REFERENCES</b>	60
	<b>APPENDICES</b>	
A	PIC Programming	63
B	Main Menu GUI Programming	65
C	Motor Control Menu GUI Programming	68
D	5V DC Motor Control GUI Programming	71
E	DC Motor Control Application Menu GUI Programming	77
F	Credit Menu GUI Programming	81
G	Abstract Menu GUI Programming	84
H	Help Menu GUI Programming	86
I	Confirm Close GUI Programming	88
J	Open and Close Port Menu GUI Programming	92
K	PIC 16F877A Data Sheet	94
L	MAX232 Data Sheet	96
M	LM35DZ Data Sheet	98
N	C1815 Data Sheet	100
O	Project circuit (hardware)	101

## LIST OF FIGURES

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
<b>1.1</b>	Block Diagram of the project	2
<b>2.1</b>	PIC	6
<b>2.2</b>	PIC16F877A pin	7
<b>2.3</b>	Part of an electric motor	15
<b>2.4</b>	LM35DZ	19
<b>2.5</b>	Darlington transistor combination circuit	21
<b>2.6</b>	Relay	22
<b>2.7</b>	MAX232 connection to DB9	23
<b>3.1</b>	Flowchart of the project	26
<b>3.2</b>	Power supply modules	27
<b>3.3</b>	Pins and signal associated with the 9-pin connector	28
<b>3.4</b>	Serial port connection to PIC	29
<b>3.5</b>	LM35 circuit	30
<b>3.6</b>	DC Motor Connection	31
<b>3.7</b>	MATLAB GUIDE Layouts	32
<b>3.8</b>	Property inspector	34
<b>3.9</b>	Example of GUI	35
<b>3.10</b>	Example M file for GUI	36
<b>3.11</b>	Initialize communication port	40
<b>3.12</b>	Open and Close of communication port	40

<b>3.13</b>	Transmit data to PIC	41
<b>3.14</b>	PIC Programming Sample	44
<b>4.1</b>	Main Menu of GUI	47
<b>4.2</b>	Credit	48
<b>4.3</b>	Abstract	49
<b>4.4</b>	Exit Button confirmations	49
<b>4.5</b>	Motor Control menu	50
<b>4.6</b>	5V DC Motor menu	51
<b>4.7</b>	Communication port status	51
<b>4.8</b>	Application of motor menu	52
<b>4.9</b>	Application 1 menu	53
<b>4.10</b>	Application 2 menu	53
<b>4.11</b>	Warning Pop up menu	54
<b>4.12</b>	Help Menu	55
<b>4.13</b>	Info Menu	55

## LIST OF TABLE

<b>TABLE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
<b>3.1</b>	Serial port pins and signal assignments	28
<b>3.2</b>	Basic MATLAB GUI component	33
<b>3.3</b>	Kind of Callback	38
<b>3.4</b>	Major Sections of the GUI M-file	39
<b>3.5</b>	Comparison of PICBasic and PICBasic Pro	42
<b>3.6</b>	List of standard baud rate	45
<b>3.7</b>	Modifier support by SERIN2 command	45
<b>5.1</b>	Approximation cost of component	58

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Overview**

The serial port was found on the back of most PC and it is extremely useful for robotics work. Variety devices are configured to communicate via a serial port.

This project is focus on designing the Graphical User Interface (GUI) through MATLAB to control the DC motor and LM35DZ using PIC. The PIC is a programmable interface devices or controller between MATLAB GUI (PV), DC motor and LM35DZ. The main contribution of this project is the interfacing of the MATLAB with PIC and Graphical User Interface (GUI).

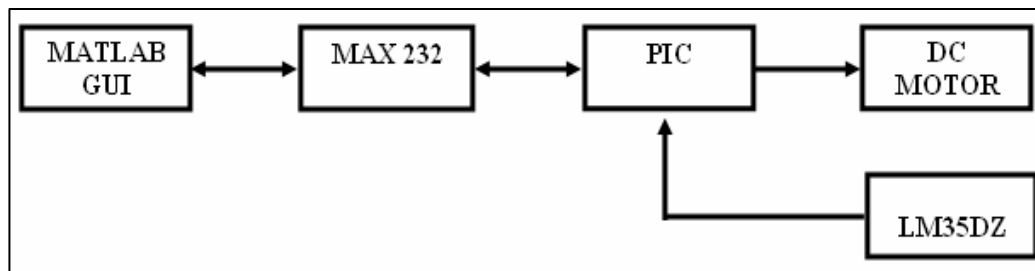
The Peripheral Interface Controller (PIC) use in this project is as controller device between Personal Computer, DC motor and LM35DZ. The PIC is use because of wide availability and economical. Beside that PIC is a free development tools and can perform many function without needed extra circuitry. The PIC also provided analog to digital converter, which is that, can be used to connect with analog temperature sensor.

The PIC program is using the PICBasic Pro Compiler. The PICBasic Pro Compiler produces code that may be programmed into a wide variety of PICmicro microcontroller.

The purpose using MATLAB in creating the GUI is because it already has Graphical User Interface Development Environment (GUIDE) that provides a set of tool for crating GUI. These tools can simplify the process of the laying out and programming GUIs.

The GUI create in MATLAB with appropriate coding will control the DC motor via serial port that interface with the PIC. There are many advantage by using the DC motor, among that the DC motor has no adverse effect on power quality and the speed is proportional to the magnetic flux.

This project is to control the DC motor application by using GUI in MATLAB and PIC controller. There are two modes to controller the DC motor application. The first mode is control the DC motor directly. For the second mode is to control the DC motor we use the analog temperature sensor to detect the current temperature and measured the value of temperature.



**Figure 1.1:** Block diagram of the project

## **1.2 Objective**

The objectives are:

- i. To develop transmitter and receiver program using MATLAB GUIDE
- ii. To control and monitoring the application of motor control system.
- iii. Able to interface software (MATLAB GUI) with hardware using serial port.

The important of this project is to interface the MATLAB GUI with the PIC. Then, the important part of this project is to receive a signal from sensor that will transmit to MATLAB GUI and interface using PIC. After that, the programming will send the signal to control the motor application automatically.

## **1.3 Scope of Project**

The scopes of this project are laying out the GUI in MATLAB GUIDE and create programming for the GUI's. Secondly Prepare the PIC circuitry and serial connections (DB9) circuit for interfacing part. For the third part is to build temperature sensor circuit and interface with PIC. And the last part is creating program for PIC using PICBasic Pro Compiler to control the DC motor.

For this project, there are two scopes. The scope of project is dividing to software part and hardware part:

For the software part, we have:

- i. MATLAB programming
- ii. PIC programming
- iii. PICBasic Pro Compiler

For the hardware part, we have:

- i. 2 ways serial parallel port (transmit and received for input and output)
- ii. PIC 16F877A
- iii. DC Motor and other components
- iv. Relay 6V
- v. LM35DZ

## **1.4 Problem Statement**

The main objective in this project is to interface the MATLAB GUI with the PIC. It is a difficult part to develop the program for MATLAB and the PIC simultaneously to make the interfacing part. By using the PicBasic Pro Compiler software to develop programming to control DC motor, it can reduce the difficulty by comprising a list of statements that are written in a programming language like assembler, C, or PBASIC. With this opportunity, the men in charge do not have to take long time to write and troubleshoot the program.



## **1.5 Thesis Organization**

This thesis consists of five chapters including this chapter. The contents of each chapter are outlined as follows;

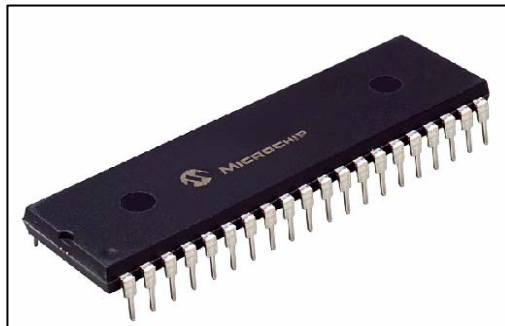
Chapter 2 contains a detailed description each part of project. It will explain about the, PIC, DC motor, temperature sensor and MATLAB GUIDE. Chapter 3 includes the project methodology. This will explain how the project is organized and the flow of the process in completing this project. Chapter 4 presents the expected result of simulation runs using MATLAB GUIDE and the analysis of the project. Finally the conclusions for this project are presented in Chapter 5.

## CHAPTER 2

### LITERATURE REVIEW

#### 2.1 PIC Microcontroller

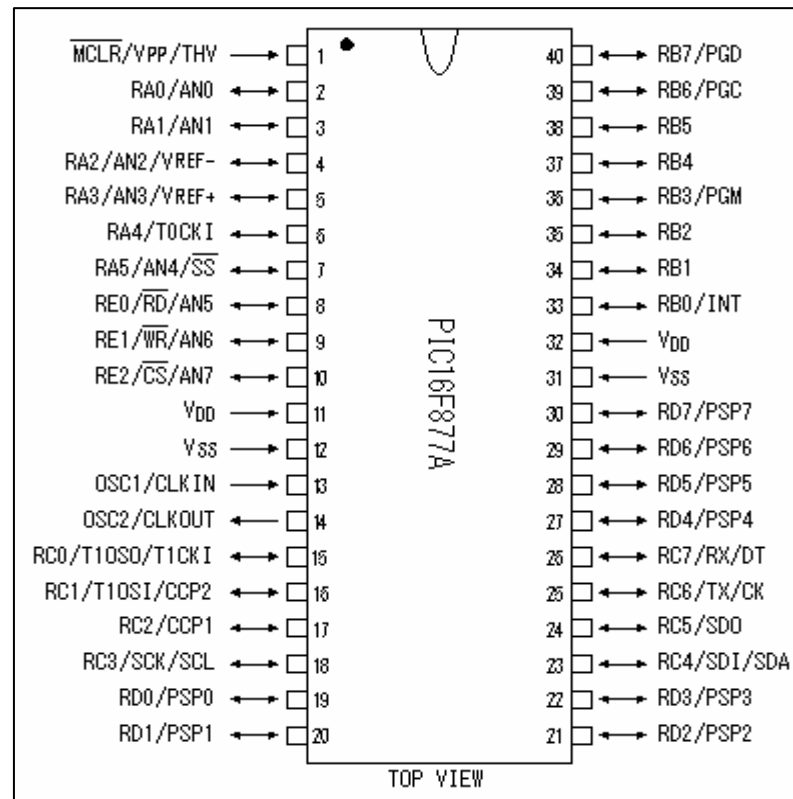
PIC is a family of Harvard architecture microcontrollers made by Microchip Technology, derived from the PIC1640 originally developed by General Instrument's Microelectronics Division. The name PIC initially referred to "Programmable Interface Controller", but shortly thereafter was renamed "Programmable Intelligent Computer" [5].



**Figure 2.1:** PIC

### 2.1.1 Origins

The original PIC was built to be used with GI's new 16-bit CPU, the CP1600. While generally a good CPU, the CP1600 had poor I/O performance, and the 8-bit PIC was developed in 1975 to improve performance of the overall system by offloading I/O tasks from the CPU. The PIC used simple microcode stored in ROM to perform its tasks, and although the term wasn't used at the time, it shares some common features with RISC designs.



**Figure 2.2:** PIC16F877A pin

In 1985 General Instruments spun off their microelectronics division, and the new ownership cancelled almost everything — which by this time was mostly out-of-

date. The PIC, however, was upgraded with EPROM to produce a programmable channel controller, and today a huge variety of PICs are available with various on-board peripherals (serial communication modules, UARTs, motor control kernels, etc.) and program memory from 512 words to 64k words and more (a "word" is one assembly language instruction, varying from 12, 14 or 16 bits depending on the specific PIC micro family).

Microchip Technology does not use PIC as an acronym; in fact the brand name is PICmicro. It is generally regarded that PIC stands for Peripheral Interface Controller, although General Instruments' original acronym for the initial PIC1640 and PIC1650 devices was "Programmable Interface Controller". The acronym was quickly replaced with "Programmable Intelligent Computer".

The Microchip 16C84 (PIC16x84), introduced in 1993[6] was the first CPU with on-board EEPROM memory. This electrically-erasable memory made it cost less than CPUs that required a quartz "erase window" for erasing EPROM.[5]

### **2.1.2 PIC Microcontroller Option**

A microcontroller (also MCU or  $\mu$ C) is a functional computer system-on-a-chip. It contains a processor core, memory, and programmable input/output peripherals. While the PIC controller chips are the combination the function of microprocessor, ROM program memory, same RAM memory and input-output interface in one single package which is economical and easy to use.

The PIC – Logicator system is design to be used to program a range of 8, 18, 28 pin reprogrammable PIC microcontroller which provide a variety of input-output, digital input and analogue input options to suit students project uses [11].

Reprogrammable “FLASH Memory” chips have been selected as the most economical for student use. If a student needs to amend to control system as the project is evaluated and developed, the chip can simply be taken out of the product and reprogrammed with an edited version of the floe sheet [11].

The PIC devices generally feature is sleep mode (power savings), watchdog timer, various crystal or RC oscillator configurations, or an external clock [5].

### **2.1.3 Variants**

Within a series, there are still many device variants depending on what hardware resources the chip features.

- General purpose I/O pins.
- Internal clock oscillators.
- 8/16 Bit Timers.
- Internal EEPROM Memory.
- Synchronous/Asynchronous Serial Interface USART.
- MSSP Peripheral for I<sup>2</sup>C and SPI Communications.
- Capture/Compare and PWM modules.
- Analog-to-digital converters (up to ~50 kHz).
- USB, Ethernet, CAN interfacing support.
- External memory interface.
- Integrated analog RF front ends (PIC16F639, and rfPIC).

- KEELOQ Rolling code encryption peripheral (encode/decode)
- And many more.

## **2.2 PIC Basic Pro Compiler**

The PICBASIC PRO™ Compiler (or PBP) is the easiest way to program the fast and powerful Microchip Technology PICmicro microcontrollers (MCUs). PICBASIC PRO converts the BASIC programs into files that can be programmed directly into PICmicro microcontrollers (MCUs). The English-like BASIC language is much easier to read and write the quirky Microchip assembly language (likes machine language and assembly language).

The PicBasic Pro Compiler instruction set is upward compatible with the BASIC Stamp II and Pro uses BS2 syntax. Programs can be compiled and programmed directly into a PICmicro MCU, eliminating the need for a BASIC Stamp module. These programs execute much faster and may be longer than their Stamp equivalents. They may also be protected so no one can copy your code [4].

The PicBasic Pro Compiler also can create programs for any of Microchip's PICmicro microcontrollers and works with most PICmicro MCU programmers, including the elbas Serial Programmer. A printed manual and sample programs are included to get you started [3].

## **2.3 Graphical User Interface (GUI)**

### **2.3.1 General Definition of GUI**

A graphical user interface (or GUI, often pronounced “gooey”), is a particular case of user interface for interacting with a computer which employs graphical images and widgets in addition to text to represent the information and actions available to the user [1][2].

Graphical user interfaces (GUIs) are being increasingly used to provide users to computer simulations with a friendly and visual approach to specifying all input parameters, thus making it easier to describe what is needed to run a program. GUI allows the user to see everything at once. Hence, data entry becomes much easier because of the visual aid instead of trying to remember all of the different command-line prompts, or text input files.

### **2.3.2 MATLAB GUI**

GUIDE, the MATLAB® Graphical User Interface development environment provides a set of tools for creating graphical user interfaces (GUIs). These tools greatly simplify the process of designing and building GUIs. User can use the GUIDE tools to lay out the GUI [12].

MATLAB GUI is also a graphical display that contains device or component which enable a user to interactive task. To this task perform, user of the GUI does not have to create a script or type commands at the command line.

By using the GUIDE Layout Editor in MATLAB, user can lay out a GUI easily by clicking and dragging GUI components. The GUI component such as panels, buttons, and text fields, sliders, menus, and so on. In MATLAB, a GUI can also display data in tabular form or as plots, and can group related component.

GUIDE in MATLAB provide the set of tools for creating graphical user interfaces (GUIs). The tools function to simplify the process of laying out and programming of GUIs.

GUIDE automatically generates an M-file that controls how the GUI operates. The M-file initializes the GUI and contains a framework for all the GUI callbacks while the commands that are executed when a user clicks a GUI component. Using the M-file editor, the user can add code to the callbacks to perform the functions needed [12].

### **2.3.3 MATLAB GUIDE**

GUIDE, the MATLAB graphical user interface development environment, provides a set of tools for creating graphical user interfaces (GUIs). These tools simplify the process of laying out and programming GUIs [13].

- i. GUIDE is primarily a set of layout tools
- ii. GUIDE also generates an M-file that contains code to handle the initialization and launching of the GUI



The M-file will provide a framework for the implementation of the callbacks, the functions that execute when users activate a component in the GUI [13].

#### **2.3.3.1 Two Basic in Process of implementing a GUI**

The two basic tasks in Process of implementing a GUI is first, laying out a GUI where MATLAB implement GUIs as figure windows containing various styles of uicontrol (User Interface) objects. The second task is programming the GUI, where each object must be program to perform the intended action when activated by the user of GUI [14].

#### **2.3.4 Operation of GUI**

Each component, and the GUI itself, is associated with one or more user-written routines known as callbacks. The execution of each callback is triggered by a particular user action such as, mouse click, pushbuttons, toggle buttons, lists, menus, text boxes selection of a menu item, or the cursor passing over a component and so forth [15].

By clicking the button triggers the execution of a callback. A mouse click or a key press is an event, and the MATLAB program must respond to each event if the program is to perform its function. For example, if a user clicks on a button, that event must cause the MATLAB code that implements the function of the button to be executed. The code executed in response to an event is known as a call back [15].

This kind of programming is often referred to as event-driven programming. The event in the example is a button click. In event-driven programming, callback execution is asynchronous, controlled by events external to the software. In the case of MATLAB GUIs, these events usually take the form of user interactions with the GUI [15].

The writer of a callback has no control over the sequence of events that leads to its execution or, when the callback does execute, what other callbacks might be running simultaneously [15].

Callbacks:

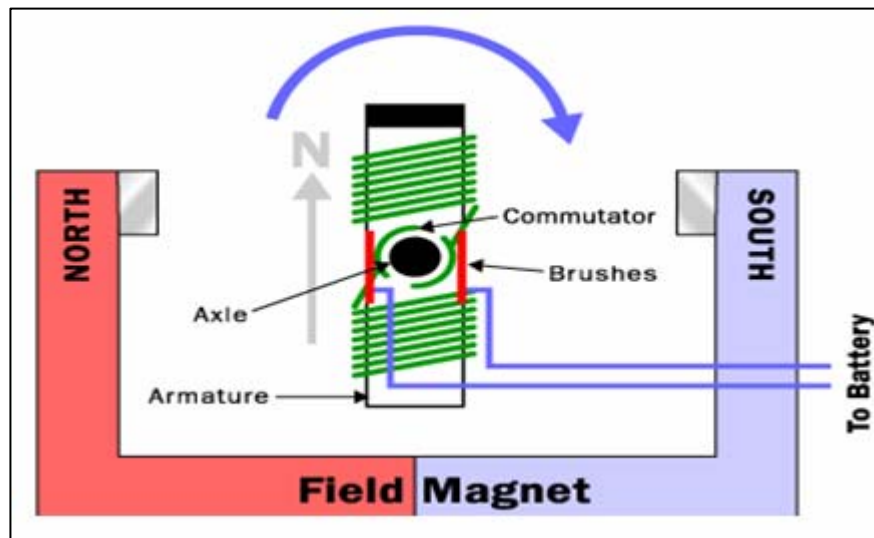
- i. Routine that executes whenever you activate the uicontrol object.
- ii. Define this routine as a string that is a valid MATLAB expression or the name of an M-file.
- iii. The expression executes in the MATLAB workspace.

## 2.4 DC Motor

### 2.4.1 Introduction

A DC motor is designed to run on DC electric power. An electric motor is a device that transforms electrical energy into mechanical energy by using the motor effect.

DC motors consist of rotor-mounted windings (armature) and stationary windings (field poles). In all DC motors, except permanent magnet motors, current must be conducted to the armature windings by passing current through carbon brushes that slide over a set of copper surfaces called a commutator, which is mounted on the rotor [6].



**Figure 2.3:** Part of an electric motor

The commutator bars are soldered to armature coils. The brush/commutator combination makes a sliding switch that energizes particular portions of the armature, based on the position of the rotor. This process creates north and south magnetic poles on the rotor that are attracted to or repelled by north and south poles on the stator, which are formed by passing direct current through the field windings. It's this magnetic attraction and repulsion that causes the rotor to rotate [6].

#### **2.4.2 The Advantage and The Drawbacks**

The advantage

- The greatest advantage of DC motors may be speed control. Since speed is directly proportional to armature voltage and inversely proportional to the magnetic flux produced by the poles, adjusting the armature voltage and/or the field current will change the rotor speed.
- Today, adjustable frequency drives can provide precise speed control for AC motors, but they do so at the expense of power quality, as the solid-state switching devices in the drives produce a rich harmonic spectrum. The DC motor has no adverse effects on power quality.

The drawback

- Power supply, initial cost, and maintenance requirements are the negatives associated with DC motors
- Rectification must be provided for any DC motors supplied from the grid. It can also cause power quality problems.

- The construction of a DC motor is considerably more complicated and expensive than that of an AC motor, primarily due to the commutator, brushes, and armature windings. An induction motor requires no commutator or brushes, and most use cast squirrel-cage rotor bars instead of true windings — two huge simplifications.

### **2.4.3 Type of DC Motor**

#### **2.4.3.1 Stepper Motor**

A stepper motor is a brushless, synchronous electric motor that can divide a full rotation into a large number of steps, for example, 200 steps. Thus the motor can be turned to a precise angle [7]. A stepper motor is an electromechanical device which converts electrical pulses into discrete mechanical movements and is a unique type of dc motor that rotates in fixed steps of a certain number of degrees. Step size can range from 0.9 to 90 degree [7] [8].

The speed of the motor shafts rotation is directly related to the frequency of the input pulses and the length of rotation is directly related to the number of input pulses applied. The motors rotation has several direct relationships to these applied input pulses. The sequence of the applied pulses is directly related to the direction of Motor shafts rotation [8] [9]. The stepper motors has an excellent response to startup, stopping and reverse [7].

There are three main of stepper motor type. First is Permanent Magnet (PM) Motors second is Variable Reluctance (VR) Motors and the third is Hybrid Motors.

#### **2.4.3.2 Coreless DC Motor**

- Optimized for rapid acceleration, these motors have a rotor that is constructed without any iron core.
- Because the rotor is much lighter in weight (mass) than a conventional rotor formed from copper windings on steel laminations, the rotor can accelerate much more rapidly, often achieving a mechanical time constant less than 1 ms.
- These motors were commonly used to drive the capstan(s) of magnetic tape drives and are still widely used in high-performance servo-controlled systems [8].

#### **2.4.3.3 Brushless DC Motor**

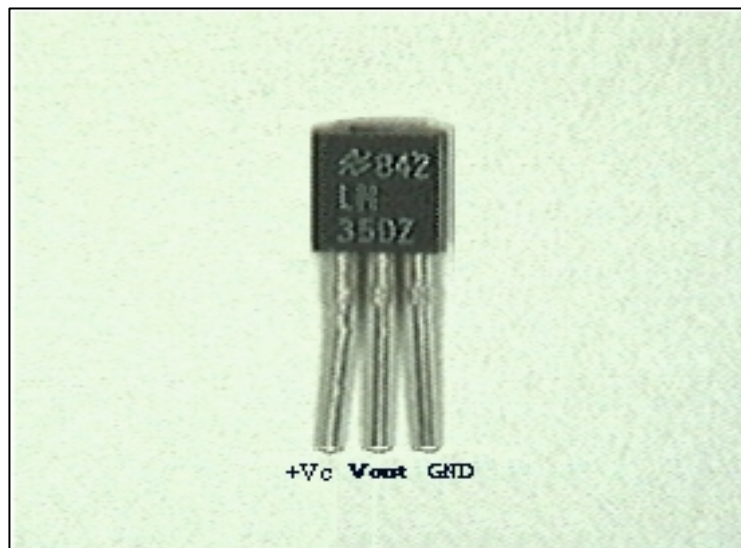
- A brushless DC motor (BLDC) is an AC synchronous electric motor that from a modeling perspective looks very similar to a DC motor.
- In a BLDC motor, the electromagnets do not move; instead, the permanent magnets rotate and the armature remains static.
- In order to do this, the brush-system/commutator assembly is replaced by an
- Intelligent electronic controller. The controller performs the same power distribution found in a brushed DC-motor, but using a solid-state circuit rather than a commutator/brush system [8].

## 2.5 Temperature Sensor

Temperature sensor is very common used in industry. It is also used in electric and electronics. Temperature sensor is important to detect the temperature level in industrial world. Many type of temperature sensor have in the industry:

- Thermistors
- Thermocouple
- LM35DZ

In this project, the LM35 will be used as the temperature sensor. The LM35 will be using in this project because it can measure temperature more accurately than a using thermistor. The sensor circuitry is sealed and not subject to oxidation. The LM35 also generates a higher output voltage than thermocouples and may not require that the output voltage be amplified [6].



**Figure 2.4:** LM35DZ

### The LM35DZ work

- It has an output voltage that is proportional to the Celsius temperature.
- The scale factor is  $.01\text{V}/^{\circ}\text{C}$ .
- The LM35 does not require any external calibration or trimming and maintains an accuracy of  $\pm 0.4^{\circ}\text{C}$  at room temperature and  $\pm 0.8^{\circ}\text{C}$  over a range of  $0^{\circ}\text{C}$  to  $+100^{\circ}\text{C}$ .
- Another important characteristic of the LM35DZ is that it draws only 60 micro amps from its supply and possesses a low self-heating capability. The sensor self-heating causes less than  $0.1^{\circ}\text{C}$  temperature rise in still air.

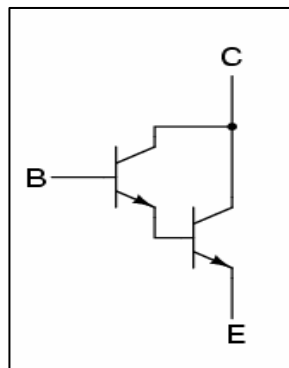
### The LM35DZ features

- Calibrated directly in  $^{\circ}\text{Celsius}$  (Centigrade)
- Linear  $+ 10.0\text{ mV}/^{\circ}\text{C}$  scale factor
- $0.5^{\circ}\text{C}$  accuracy guaranteeable (at  $+25^{\circ}\text{C}$ )
- Rated for full  $-55^{\circ}$  to  $+150^{\circ}\text{C}$  range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts • Less than  $60\text{ }\mu\text{A}$  current drain
- Low self-heating,  $0.08^{\circ}\text{C}$  in still air • Nonlinearity only  $\pm 1/4^{\circ}\text{C}$  typical
- Low impedance output,  $0.1\text{ Ohm}$  for  $1\text{ mA}$  load



## 2.6 Darlington Transistor C1815

Darlington transistor is a semiconductor device which combines two bipolar junction transistors in a single device. So that the current amplified by the first is amplified further by the second. This configuration gives a high current gain (written  $\beta$ ,  $h_{fe}$ , or  $h_{FE}$ ) and can take less space than two separate transistors because the two transistors can use a shared collector. Integrated circuit packages are available, but it is still common also to use two separate transistors [16].



**Figure 2.5:** Darlington transistor combination circuit

## 2.7 Relay

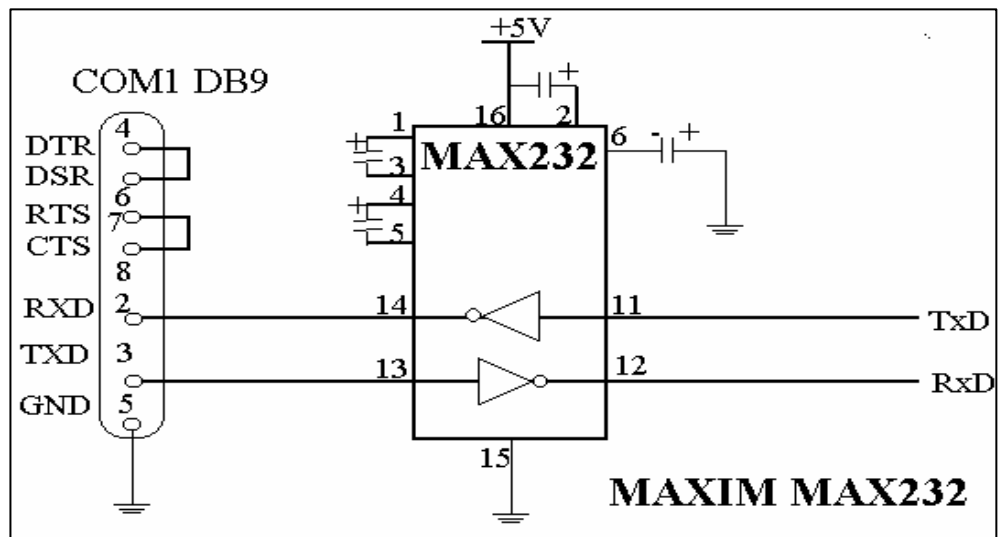
Relay is an electrical switch that opens and closes under the control of another electrical circuit. The switch is operated by an electromagnet to open or close one or many sets of contacts [17]. The coil current can be on or off so relays have two switch positions and they are double throw (changeover) switches. Relays allow one circuit to switch a second circuit which can be completely separate from the first.



**Figure 2.6:** Relay

## 2.8 MAX232

The MAX232 is an integrated circuit that converts signals from an RS-232 serial port to signals suitable for use in TTL compatible digital logic circuits. The MAX232 is a dual driver/receiver and typically converts the RX, TX, CTS and RTS signals. The drivers provide RS-232 voltage level outputs (approx.  $\pm 7.5$  V) from a single +5 V supply via on-chip charge pumps and external capacitors. This makes it useful for implementing RS-232 in devices that otherwise do not need any voltages outside the 0 V to +5 V range, as power supply design does not need to be made more complicated just for driving the RS-232 in this case. The receivers reduce RS-232 inputs (which may be as high as  $\pm 25$  V), to standard 5 V TTL levels. These receivers have a typical threshold of 1.3 V, and a typical hysteresis of 0.5 V [18].



**Figure 2.7:** MAX232 connection to DB9

## **CHAPTER 3**

### **METHODOLOGY**

#### **3.1 Introduction**

This chapter explains about what is the method that has been used to complete this project. It describes on how the project is organized and the flow of the steps in order to complete this project. The methodology is consoled of two parts, which is developing the hardware to interface with MATLAB. The other part is developing the programming for MATLAB and the PIC to control DC motor application.